

Refine Search

Search Results -

Term	Documents
709/\$	0
709/200	621
709/201	1040
709/202	691
709/203	2245
709/204	614
709/205	382
709/206	931
709/207	275
709/208	349
709/209	134
(709/\$.CCLS. AND (MAP\$ WITH INTERMEDIATES\$ WITH ADDRESS\$)).USPT.	14

There are more results than shown above. [Click here to view the entire set.](#)

Database:

☐ US Pre-Grant Publication Full-Text Database
☒ US Patents Full-Text Database
☐ US OCR Full-Text Database
☐ EPO Abstracts Database
☐ JPO Abstracts Database
☐ Derwent World Patents Index
☐ IBM Technical Disclosure Bulletins

Search:

L3

Refine Search

Recall Text 

Clear

Interrupt

Search History

DATE: Monday, July 26, 2004 [Printable Copy](#) [Create Case](#)

Set Name Query

side by side

DB=USPT; PLUR=YES; OP=ADJ

Hit Count Set Name

result set

<u>L3</u>	709/\$.ccls. and (map\$ with intermediate\$ with address\$)	14	<u>L3</u>
<u>L2</u>	L1 and (map\$ with intermediate\$ with address\$)	0	<u>L2</u>
<u>L1</u>	(6738821 or 6061723).pn.	2	<u>L1</u>

END OF SEARCH HISTORY

[Previous Doc](#) [Next Doc](#) [Go to Doc#](#)
[First Hit](#) [Fwd Refs](#)

☐ **Generate Collection**

L3: Entry 4 of 14

File: USPT

Apr 10, 2001

DOCUMENT-IDENTIFIER: US 6216168 B1

TITLE: Perspective-based shared scope address resolution method and apparatus

Detailed Description Text (23):

The above request is implemented, as shown in FIG. 6, in the hierarchical network 600 including scopes 602-608. As above, each of these scopes includes at least one address mapping. Starting in "switch5" scope 602 there is no mapping of the address (a, a1) to a type "d" in "switch5" scope 602. There is, however, a mapping to the address tuple (b, b1). In step 610 address (a, a1) is resolved to address (b, b1) which becomes an "intermediate" lhs tuple. The process goes up in the hierarchy when it cannot resolve any further in a particular scope. Step 612 moves to the next higher scope, "3rdfl" scope 604, where there is a mapping of the intermediate tuple (b, b1) to the address (c, c1). So, in step 614, address (b, b1) is resolved to tuple (c, c1) and this becomes the "intermediate" lhs tuple. At this point, however, the search goes back, step 616, to the original specific scope, i.e., "switch5" to resolve the intermediate address (c, c1) in order to find an address with type "d". Note that the resolution does not continue from (c, c1) to (d, d1.in3rdfl). There is only one resolution per scope. The process returns to the original scope upon any partial resolution in a higher scope (rather than continuing along the series of the more general higher scope). As shown in "switch5" scope 602 there is a mapping of (c, c1) to (d, d1.insw5). In step 618, intermediate tuple (c, c1) is resolved to tuple (d, d1.insw5). As a final result, the address (a, a1) resolves to a "d" type with a value of "d1.insw5" in scope "switch5".

Detailed Description Text (25):

As shown in FIG. 7, the hierarchical directory (tree structure) 400 is the same as that of FIG. 4. Starting in "switch5" scope 408 there is no mapping of the address (a, a1) to an address with the type "e". Step 702 resolves address (a, a1) to address (b, b1) as an intermediate lhs tuple. Step 704 then moves to the next higher scope, "3rdfl" scope 406. The tuple (b, b1) is intermediately resolved to an address (c, c1) in the "3rdfl" scope 406 in step 706. Since the method always tries to resolve at the lowest scope level, step 708 moves back to scope 408. The intermediate address (c, c1) is checked in the "switch5" node 408 where a resolution, step 710, from address (c, c1) to address (d, d1.insw5) is performed. Since the desired type "e" is still not found, the search moves up to the "3rdfl" node 406, step 712, where there is also no mapping of the intermediate address (d, d1.insw5) to an address with a type of "e". Moving up to the next higher hierarchical "durham" scope 404, step 714, there is no mapping for the intermediate address (d, d1.insw5). There is, however, a mapping for the address (d, d1). As between the address (d, d1.insw5) and the address (d, d1), they have the same type, i.e., "d", and the values have the same prefix, i.e., "d1". A mapping of address (d, d1) to (e, e1) is present in "durham" scope 404. In step 716, intermediate address (d, d1) is resolved to address (e, e1) based on the common "d1" prefix. As a result, the address of (a, a1) in scope "switch5" resolves to an address (e, e1) in the "durham" scope.

Detailed Description Text (27):

As shown in FIG. 8, the above resolution request is implemented in the same

hierarchical directory (tree structure) 400 as found in FIG. 4. Starting in the "switch5" scope 408, the method looks for a mapping of address (a, a1) to an address tuple having a type of "f". There is no such mapping, although there is a mapping of the (a, a1) address to an address (b, b1). As a result, in step 802, an intermediate resolution resolves the address (a, a1) to intermediate address (b, b1). In step 804, the next higher level of hierarchy, i.e., "3rdf1" scope 406, is checked to see if there is a mapping from address (b, b1) to an address having a type of "f". Since such a mapping does not exist, step 806 intermediately resolves the address (b, b1) to the intermediate address (c, c1) and step 808 returns to the "switch5" scope 408.

Detailed Description Text (28):

There is no mapping for the address (c, c1) to the desired type, so in step 810 an intermediate resolution of the address (c, c1) to the address (d, d1.insw5) is performed. Step 812 moves to the next higher "3rdf1" scope 406 where there is no mapping of address (d, d1.insw5), so step 814 moves to the next higher "durham" scope 404. At this scope level, there is a mapping of the address (d, d1) to the address (e, e1). Relying on the common prefix between "d1.insw5" and "d1", step 816 intermediately resolves to the address (e, e1).

Current US Original Classification (1):

709/245

Current US Cross Reference Classification (1):

709/206

Current US Cross Reference Classification (2):

709/230

CLAIMS:

2. The system as recited in claim 1, wherein the plurality of executable methods further comprises:

when the first address type and value cannot be resolved to the target address type and value in the first node, a method for attempting to resolve the first address type and value to a first intermediate address type and value as a function of the address mapping elements in the first node; and

a method for attempting to resolve the first intermediate address type and value to the target type and value as a function of the address mapping elements in a second node at a second hierarchical level, higher than the first hierarchical level.

3. The system as recited in claim 2, wherein the plurality of executable methods further comprises:

when the first intermediate address type and value cannot be resolved to the target address type and value in the second node, a method for attempting to resolve the first intermediate address type and value to a second intermediate address type and value as a function of the address mapping elements in the second node; and

a method for attempting to resolve the second intermediate address type and value to the target address type and value as a function of the address mapping elements in the first node.

4. The system as recited in claim 3, wherein the plurality of executable methods further comprises:

when the second intermediate address type and value cannot be resolved to the target address type and value in the first node, a method for attempting to resolve

the second intermediate address type and value to a third intermediate address type and value as a function of the address mapping elements in the first node; and

a method for attempting to resolve the third intermediate address type and value to the target address type and value as a function of the address mapping elements in the second node.

5. The system as recited in claim 4, wherein the plurality of executable methods further comprises:

when the third intermediate address type and value cannot be resolved to the target address type and value in the second node, a method for attempting to resolve the third intermediate address type and value to the target address type and value as a function of the address mapping elements in a third node at a third hierarchical level, the third hierarchical level being higher than the first and second hierarchical levels.

6. The system as recited in claim 5, wherein the method for attempting to resolve the third intermediate address type and value to the target address type and value as a function of the address mapping elements in the third node comprises:

a method for attempting to resolve the third intermediate address type and value to a fourth intermediate address type and value where the third intermediate value and the fourth intermediate value have a common prefix; and

a method for attempting to resolve the fourth intermediate address type and value to the target address type and value.

8. The system as recited in claim 1, wherein the plurality of executable methods further comprises:

a method for attempting to resolve the first address type and value to a first intermediate address type and value as a function of the address mapping elements in the first node when the first address type and value cannot be resolved to the target address type and value as a function of the address mapping elements in the first node; and

a method for attempting to resolve the first intermediate address type and value to the target address type and value;

wherein the first intermediate address value and the target address value have a common prefix.

13. A method of resolving a starting address type and value to a target address type and value in a system including a hierarchical directory including a tree structure, the tree structure including at least a starting hierarchical node at a starting hierarchical level, each hierarchical node including at least one address mapping set, and each hierarchical node representing a respective hierarchical level within the system, the method comprising the steps of:

(a) attempting to resolve the starting address type and value to the target address type and value as a function of the address mapping set in the starting hierarchical node;

(b) when the starting address type and value cannot be resolved to the target address type in step (a), performing one of the following steps:

(c) resolving the starting address type and value to a first intermediate address type and value as a function of the address mapping set in the starting node; and

(d) attempting to resolve the starting address type and value to the target address type and value as a function of the address mapping set in a second hierarchical node at a second hierarchical level higher than the first hierarchical level.

14. The method as recited in claim 13, wherein:

(e) when the starting address type and value have been resolved to the first intermediate address type and value in step (c), attempting to resolve the first intermediate address type and value to the target address type and value as a function of the address mapping set in a second hierarchical node at a second hierarchical level higher than the starting hierarchical level; and

(f) when the first intermediate address type and value cannot be resolved to the target address type and value in step (e), performing one of the following steps:

(g) resolving the first intermediate address type and value to a second intermediate address type and value as a function of the address mapping set in the second hierarchical node; and

(h) attempting to resolve the first intermediate address type and value to the target address type and value as a function of the address mapping set in a third hierarchical node at a third hierarchical level higher than the second hierarchical node.

15. The method as recited in claim 14, wherein:

when the first intermediate address type and value have been resolved to the second intermediate address type and value in step (g), attempting to resolve the second intermediate address type and value to the target address type and value as a function of the address mapping set in the starting hierarchical node.

18. The apparatus as recited in claim 17, further comprising:

third means for mapping the first address type and value to a first intermediate address type and value when the second means for mapping cannot map the first address type and value to the second address type and value; and

means for mapping the first intermediate address type and value to the second address type and value.

20. The computer-readable medium as recited in claim 19, wherein the method further comprises:

if the first address type and value cannot be resolved to the target address type and value in the first node, attempting to resolve the first address type and value to a first intermediate address type and value as a function of the address mapping elements in the first node; and

attempting to resolve the first intermediate address type and value to the target type and value as a function of the address mapping elements in a second node at a second hierarchical level, the second hierarchical level being higher than the first hierarchical level.

21. The computer-readable medium as recited in claim 20, wherein the method further comprises:

if the first intermediate address type and value cannot be resolved to the target address type and value in the second node, attempting to resolve the first intermediate address type and value to a second intermediate address type and value as a function of the address mapping elements in the second node; and

attempting to resolve the second intermediate address type and value to the target address type and value as a function of the address mapping elements in the first node.

22. The computer-readable medium as recited in claim 21, wherein the method further comprises:

if the second intermediate address type and value cannot be resolved to the target address type and value in the first node, attempting to resolve the second intermediate address type and value to a third intermediate address type and value as a function of the address mapping elements in the first node; and

attempting to resolve the third intermediate address type and value to the target address type and value as a function of the address mapping elements in the second node.

23. The computer-readable medium as recited in claim 22, wherein the method further comprises:

if the third intermediate address type and value cannot be resolved to the target address type and value in the second node, attempting to resolve the third intermediate address type and value to the target address type and value as a function of the address mapping elements in a third node at a third hierarchical level, the third hierarchical level being higher than the first and second hierarchical levels.

24. The computer-readable medium as recited in claim 23, wherein the attempting to resolve the third intermediate address type and value to the target address type and value as a function of the address mapping elements in the third node comprises:

attempting to resolve the third intermediate address type and value to a fourth intermediate address type and value where the third intermediate value and the fourth intermediate value have a common prefix; and

attempting to resolve the fourth intermediate address type and value to the target address type and value.

26. The computer-readable medium as recited in claim 19, wherein the method further comprises:

attempting to resolve the first address type and value to a first intermediate address type and value as a function of the address mapping elements in the first node if the first address type and value cannot be resolved to the target address type and value as a function of the address mapping elements in the first node; and

attempting to resolve the first intermediate address type and value to the target address type and value;

wherein the first intermediate address value and the target address value have a common prefix.

[Previous Doc](#) [Next Doc](#) [Go to Doc#](#)